

dunetpc - Task #23411

Create a new package/repository for protodune analysis

10/11/2019 01:11 PM - Tingjun Yang

Status:	Closed	Start date:	10/11/2019
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			
Description			
We plan to create a new package/repository for protodune analysis. This new package will depends on dunetpc.			
We will create a new redmine project first. Tom and I agreed on the name protoduneana.			

History

#1 - 10/11/2019 03:37 PM - Thomas Junk

The redmine project protoduneana has been created and a repository of that name has been requested. it'll take a few minutes for the cron job to get around to making the repo. I have added Tingjun, Leigh, David, Gavin, and myself as managers. I will add the larsoft_users list (the hundreds-long list of developers of larsoft and dunetpc) as developers later when we are happy to let others push to it.

It is a subproject of dunetpc, following the larsoft and MicroBooNE examples.

Note to self -- we also need to update the Jenkins build script.

#2 - 10/14/2019 09:16 AM - Tingjun Yang

I added a CMakeLists.txt file and ups directory. However, when I tried to build it with dunetpc, I got the following error:

```
Determining if the pthread_create exist failed with the following output:
Change Dir: /data/tjyang/dune/larsoft/build_slf7.x86_64/CMakeFiles/CMakeTmp
```

```
Run Build Command(s) :/cvmfs/larsoft.opensciencegrid.org/products/ninja/v1_8_2/Linux64bit+3.10-2.17/bin/ninja cmTC_f5159
[1/2] Building C object CMakeFiles/cmTC_f5159.dir/CheckSymbolExists.c.o
[2/2] Linking C executable cmTC_f5159
FAILED: cmTC_f5159
: && /cvmfs/larsoft.opensciencegrid.org/products/gcc/v7_3_0/Linux64bit+3.10-2.17/bin/gcc -rdynamic CMakeFiles/cmTC_f5159.dir/CheckSymbolExists.c.o -o cmTC_f5159
&& :
CMakeFiles/cmTC_f5159.dir/CheckSymbolExists.c.o: In function `main':
CheckSymbolExists.c:(.text+0x16): undefined reference to `pthread_create'
collect2: error: ld returned 1 exit status
ninja: build stopped: subcommand failed.
```

#3 - 10/15/2019 09:25 AM - Thomas Junk

I found the repo had no master branch and executing mrbg produced a complaint about HEAD pointing to a nonexistent ref, even though .git/HEAD pointed to the develop branch. I created and pushed a master branch.

I also added cetbuildtools to the product_deps file and now the package builds without complaining about pthread.

#4 - 10/16/2019 09:51 PM - Tingjun Yang

Thomas Junk wrote:

I found the repo had no master branch and executing mrbg produced a complaint about HEAD pointing to a nonexistent ref, even though .git/HEAD pointed to the develop branch. I created and pushed a master branch.

I also added cetbuildtools to the product_deps file and now the package builds without complaining about pthread.

I have both dunetpc and protoduneana in my srcs directory and I still get the pthread error.

#5 - 10/16/2019 10:06 PM - Thomas Junk

This script works for me:

```
#!/bin/sh
```

```
LARSOFT_VERSION=v08_32_01  
COMPILER=e17  
DIRECTORY=protoduneanatest  
USERNAME=`whoami`
```

```
source /cvmfs/dune.opensciencegrid.org/products/dune/setup_dune.sh
```

```
cd /dune/app/users/${USERNAME}  
touch ${DIRECTORY}  
rm -rf ${DIRECTORY}  
mkdir ${DIRECTORY}  
cd ${DIRECTORY}  
setup larsoft ${LARSOFT_VERSION} -q ${COMPILER}:prof  
mrb newDev -q ${COMPILER}:prof  
source /dune/app/users/${USERNAME}/${DIRECTORY}/localProducts*/setup  
mkdir work  
cd srcs
```

```
mrb g -d protoduneana protoduneana
```

```
cd $MRB_BUILDDIR  
mrbsetenv  
mrb i -j4
```

#6 - 10/16/2019 10:07 PM - Thomas Junk

I did find some things missing in the CMakeLists.txt file while trying to set it up. I already started down the road of splitting things. We should talk as to not duplicate effort.

#7 - 10/16/2019 10:19 PM - Tingjun Yang

I took a look files at the dunetpc/dune/Protodune and I think we need to keep all producer modules and data product definitions in dunetpc and only move analyzer modules to protoduneana. I think we need to at least keep the following directories:

```
singlephase/CRT  
singlephase/CTB  
singlephase/DetectorServices  
singlephase/fcl  
singlephase/RawDecoding  
singlephase/Tool  
singlephase/NearlineMonitor (?)  
dualphase/fcl  
dualphase/RawDecoding  
dualphase/test
```

#8 - 10/17/2019 11:22 AM - Thomas Junk

I am testing a private build with the contents of Protodune/Analysis, PhysiscWeek, and TutorialExamples moved to the new package -- I had to change some header paths. The T0Reco directory, not under Protodune, includes some protoduneana headers. Will think about what to do there. Is T0Reco just ProtoDUNE now, or is it also intended for FD?

#9 - 10/17/2019 02:49 PM - Thomas Junk

I set a deadline of next Weds, Oct 24, for analyzers to merge their feature branches.

#10 - 10/18/2019 04:35 AM - Leigh Whitehead

About the T0Reco directory, I'm not 100% sure, but I think the modules from Josh can happily move to the protoduneana repository. I think that in the FD we'll be using light and beam triggers so we should have a T0 for everything from just the trigger as the event rate is much lower?

#11 - 10/18/2019 08:39 AM - David Adams

Could we commit some code to the new package and add a new jenkins project to build the package so we can test the latter out in advance of next weeks release? We may later want project to build multiple packages but let's start with projects for the individual packages.

I propose to build dunetpc v08_32_02_01 with the code removed and protoduneana v08_32_02_01 adding the new code. The moved code should be something no one will miss or for which interested parties are willing to move right away to protoduneana. This is because that code will be gone from dunetpc develop.

#12 - 10/29/2019 12:19 PM - David Adams

I see ProtoDUNETriggerFilter.fcl is gone from dunetpc. Presumably it and the supporting module code have moved to protoduneana.

I think we want to keep it in dunetpc so it can be used for dataprep studies and in DQM.

Same for EventFilter.

#13 - 10/29/2019 12:47 PM - Thomas Junk

Yes, that's the case -- the trigger filter moved to protoduneana. It included a header file `protodune/Analysis/ProtoDUNEDDataUtils.h`.

#14 - 10/29/2019 12:48 PM - Thomas Junk

DQM (Maxim's thing?) should now run from protoduneana as the top-level product

#15 - 10/29/2019 12:51 PM - David Adams

I mean the parts of DQM that do not require high-level reco or physics analysis code. I would prefer to keep these low-level filters down in dunetpc.

#16 - 10/29/2019 01:06 PM - Thomas Junk

We'd have to move `protoduneana/Analysis/ProtoDUNEDDataUtils.cxx`, `.h`, and `.fcl` back over to dunetpc as well. If you have a DQM job or a dataprep job that specifically depends on ProtoDUNE features, then it's not crazy to put it in protoduneana.

#17 - 10/29/2019 01:23 PM - David Adams

I want to be able to select on event number and trigger index. If we are splitting off our high-level analysis code, I think it is a little "crazy" to say our low level processing needs to depend on that code.

#18 - 10/29/2019 01:27 PM - Thomas Junk

I can put those filters and associated code into `dunetpc/dune/Protodune/singlephase/DataUtils` if you think it's better there. They really are single-phase specific. I was hoping the eventual split structure would diagonalize along detectors rather than what's high-level or low-level, but dunetpc has lots of code for several detectors stirred together. 35t, the protodunes, FD. I was hoping to take the 35t-specific stuff out but would break a lot of tests.

#19 - 10/29/2019 01:50 PM - David Adams

You raise many points here.

I would like to leave the two filters in dunetpc for now.

I think we should split along multiple dimensions: detectors, processing level, algorithms etc.

I think we want some low-level processing and want low-level filters there but I agree not want these low-level packages to depend on high-level, detector-specific constructs.

We can look at the filters and figure out how to split them and then chain them together for processing a different levels.

#20 - 10/29/2019 01:58 PM - Thomas Junk

Looks like `DuneEventFilter_module.cc` isn't specifically a protodune thing, though it was in `Protodune/singlephase`. Mind if I put it in `dunetpc/dune/EventFilters`? The stuff in there is old 35t stuff and I can make that go away.

#21 - 10/29/2019 02:33 PM - Thomas Junk

Okay, I'm working on moving those filters. Turns out lots of the analysis code used the `DataUtils` code that has to move to dunetpc, but there is a unique solution. `DuneEventFilter_module.cc` seems not to have an associated `fcl` file.

#22 - 10/29/2019 03:45 PM - Thomas Junk

I have a version with these filters:

`ProtoDUNEFembFilter_module.cc` `ProtoDUNETriggerFilter_module.cc` `ProtoDUNEUnstableHVFilter_module.cc`

in a new directory under `dunetpc/dune/Protodune/singlephase/DataUtils`

and moved the `ProtoDUNEDDataUtils.cxx`, `.h`, `.fcl` there too, and pointed all the other modules in protoduneana to the new location.

I also moved `DuneEventFilter_module.cc` to `dunetpc/dune/EventFilters` (no `fcl` file could be found for it), and removed the old 35t stuff that was in that directory, and things that depended on it (35t nearline monitor).

Time to push? We should check this all builds again. Builds for me in a test release on SL7.

#23 - 10/29/2019 04:00 PM - David Adams

Yes, if it builds, please push. I will test for my jobs. If all is OK, I can make a new release. --da

#24 - 10/29/2019 04:03 PM - Thomas Junk

Okay, pushed. I will try building from the repos.

#25 - 10/29/2019 04:48 PM - Thomas Junk

My sl6 and sl7 builds which build dunetpc and protoduneana together succeeded.

#26 - 10/30/2019 07:37 AM - David Adams

I will make a new dunetpc release: v08_33_00_02.

#27 - 10/30/2019 09:04 AM - David Adams

Christoph reports that reco CI tests for dunetpc are failing because t0reco.fcl and BeamEvent.fcl are missing in dunetpc.

Do we agree this should go back into dunetpc?

#28 - 10/30/2019 09:20 AM - Thomas Junk

I can move some components back from protoduneana to dunetpc today if the desire is for dunetpc to be the top-level product for production. But then there will be less left after much of the code been taken out of it (some modules look more standalone and can stay).

Splitting code that lives "underneath" other pieces may be easier if the desire is to maintain the functionality of dunetpc. I was thinking that if dunetpc is split up into things that go up the tree and other things that go down, we need an umbrella ups product that sets up everything, if it is not to be dunetpc itself.

#29 - 10/30/2019 09:41 AM - David Adams

I agree we will eventually want one or more umbrella "products" (might be ups). Maybe one of these is "all DUNE code" but it will be useful to have some that have narrower scope, e.g. protoDUNE SP reconstruction. That is a natural place to maintain and run many CI tests.

Long term, I think everything in dunetpc moves into other packages of limited scope with well defined dependencies.

I envision this splitting to be mostly horizontal by reco/analysis stage and by detector.

For now, I would say that protoDUNE reco stays with dunetpc rather than moves to protoduneana.

da

#30 - 10/30/2019 09:51 AM - Thomas Junk

It looks as if the boundary between analysis and reco has evolved over time, which is why code (pre-split) in dunetpc/dune/Protodune/Analysis is largely reco, and a lot of code in dunetpc/dune/Protodune/singlephase is analysis.

#31 - 10/30/2019 06:44 PM - Thomas Junk

I have versions of protoduneana and dunetpc source trees with T0Reco and BeamEvent_module.cc (and fcl), as well as the utilities that used to be in Protodune/Analysis (that's where BeamEvent_module.cc came from) moved to dunetpc, and that build on SL7. Shall I commit and push to develop?

#32 - 10/31/2019 07:58 AM - David Adams

Yes, please push so we can try these out.

#33 - 10/31/2019 10:35 AM - Christoph Alt

FYI: I just bumped the larsoft version in dunetpc develop so that the CI builds do not fail with version conflicts.

#34 - 10/31/2019 12:51 PM - Thomas Junk

Pushed dunetpc and protoduneana and built on SL7. SL6 build is running now. I put T0Reco and BeamEvent_module.cc, fcl and the utilities that were in protoduneana/Analysis into dunetpc/dune/Protodune/singlephase/DataUtils. Some of those utilities are geared towards processing MC in that they look at MC truth, while others look like they can run in production on data or MC, so I moved them all. There now no longer is a protoduneana/Analysis directory -- the name was redundant anyway. I moved the Pion and EMTaskForce directories to protoduneana/singlephase, as they are single-phase tools anyway. There were some commits that I merged and moved them again. I mentioned at today's analysis meeting that there would be a little residual instability while things are moving, and George just asked that I keep people up to speed.

#35 - 10/31/2019 02:20 PM - Thomas Junk

SL6 build succeeded after small mods to makefiles. I will re-test SL7.

#36 - 11/01/2019 04:24 AM - Christoph Alt

All CI builds and tests, including SL7, were successful. Shall we tag dunetpc v08_34_00?

#37 - 11/01/2019 07:43 AM - David Adams

Great. Yes, let's go with the new release.

#38 - 11/05/2019 11:38 AM - Tingjun Yang

Starting from v08_34_00, we could not run the following ProtoDUNE simulation job:

```
lar -c mcl12_gen_protoDune_beam_cosmics_p1GeV.fcl
%MSG-i MF_INIT_OK: Early 05-Nov-2019 11:36:39 CST JobSetup
Messagelogger initialization complete.
%MSG
%MSG-s ArtException: Early 05-Nov-2019 11:36:40 CST JobSetup
cet::exception caught in art
---- Configuration BEGIN
  The following were encountered while processing the module configurations:
    ERROR: Configuration of module with label generator encountered the following error:
    ---- Configuration BEGIN
      Library specification "ProtoDUNEBeam": does not correspond to any library in CET_PLUGIN_PATH of type "module"
    ---- Configuration END
  ---- Configuration END
%MSG
Art has completed and will exit with status 9.
```

#39 - 11/05/2019 12:51 PM - Thomas Junk

Yes, that module got moved to protoduneana as it included header files that had been moved to protoduneana, but it looks like all of those had been moved back to dunetpc so there is no barrier to moving this module back to dunetpc as well. There's also

```
ProtoDUNEBeamTPCMatching.fcl
ProtoDUNEBeamTPCMatching_module.cc
```

in the same directory -- should those be moved as well?

One can set up protoduneana v08_34_00 -q e17:prof and see if the MC generation works. On Tingjun's suggestion, I will move the ProtoDUNEBeam_module.cc back to dunetpc and leave the TPCMMatching module in protoduneana as it is an analyzer, not a producer.

#40 - 11/05/2019 02:52 PM - Thomas Junk

ProtoDUNEBeam_module.cc is now back in dunetpc/EventGenerator in develop, which is where its fcl file was all along. I removed it from protoduneana. Someone checking out dunetpc in develop and building it without updating protoduneana may have two of these modules until they pull protoduneana and do a fresh build.

#41 - 01/18/2021 01:02 PM - David Adams

- Status changed from New to Closed

This was completed long ago.